### **Практическая работа: Моделирование процесса "Заявка на отпуск"**

**Цель практики**

* Научиться анализировать текстовое описание бизнес-процесса и выделять ключевые шаги, роли и правила.
* Создать диаграмму вариантов использования (Use Case Diagram) для определения границ системы и действующих лиц.
* Построить диаграмму деятельности (Activity Diagram) для визуализации пошагового рабочего процесса.
* Разработать диаграмму классов (Class Diagram) для описания структуры данных.
* Применить на практике отображение бизнес-правил и условий в UML-моделях.

**Условие**

Вы — бизнес-аналитик в компании, которая решила автоматизировать процесс подачи и согласования заявлений на отпуск. Раньше всё делалось по электронной почте и в устных договорённостях, что приводило к путанице.

Вам необходимо описать новый, автоматизированный процесс с помощью UML-диаграмм для последующей передачи задачи команде разработчиков.

**Описание сценария**

1. **Сотрудник** заходит в корпоративную систему под своей учетной записью.
2. Он открывает форму создания заявки на отпуск и выбирает желаемые даты начала и окончания.
3. **Система** автоматически проверяет, достаточно ли у сотрудника накопленных дней отпуска. Если дней не хватает, система выдает ошибку, и процесс прерывается.
4. Если дней достаточно, заявка создается в статусе «На согласовании» и автоматически направляется **прямому руководителю** сотрудника.
5. **Руководитель** получает уведомление. Он открывает заявку и видит даты. Он может либо **одобрить**, либо **отклонить** заявку.
6. Если руководитель **отклоняет** заявку, он обязан указать причину. Заявка получает статус «Отклонена». Сотрудник получает уведомление с причиной отказа. На этом процесс завершается.
7. Если руководитель **одобряет** заявку, она получает статус «Одобрена».
8. После одобрения заявка автоматически направляется в **HR-отдел**.
9. **Сотрудник HR-отдела** обрабатывает одобренную заявку: вносит информацию в общую систему кадрового учета и обновляет график отпусков компании.
10. После обработки заявка получает финальный статус «Оформлена». Сотрудник и руководитель получают уведомление об успешном оформлении отпуска. Процесс завершен.

**Задание**

На основе этого сценария подготовьте три основные UML-диаграммы:

1. **Use Case Diagram**
2. **Activity Diagram**
3. **Class Diagram**

### **Выполнение работы**

#### **1. Use Case Diagram (Диаграмма вариантов использования)**

Эта диаграмма показывает, кто и для каких целей взаимодействует с системой.

* **Actors (Действующие лица):**
  + Сотрудник
  + Руководитель
  + HR-специалист
* **System Boundary (Границы системы):**
  + Система управления отпусками
* **Use Cases (Варианты использования):**
  + **Подать заявку на отпуск:** Основная цель Сотрудника.
    - Включает в себя (<<include>>) вариант использования **Проверить баланс дней**, который выполняет система.
  + **Согласовать заявку на отпуск:** Основная цель Руководителя. Он может одобрить или отклонить заявку.
  + **Обработать одобренную заявку:** Основная цель HR-специалиста.
  + **Просмотреть статус заявки:** Вариант использования, доступный Сотруднику и Руководителю.

#### **2. Activity Diagram (Диаграмма деятельности)**

Эта диаграмма наглядно показывает последовательность шагов и логику процесса.

* **Swimlanes (Дорожки):**
  + Сотрудник
  + Система
  + Руководитель
  + HR-специалист
* **Описание процесса:**
  + **Начальное событие** в дорожке «Сотрудник».
  + **Действие:** «Заполнить форму заявки».
  + Переход в дорожку «Система».
  + **Условие/Шлюз (Gateway):** «Достаточно дней отпуска?»
    - **[Нет]** -> **Действие:** «Показать сообщение об ошибке» -> **Конечное событие**.
    - **[Да]** -> **Действие:** «Создать заявку в статусе 'На согласовании'».
  + Переход в дорожку «Руководитель».
  + **Действие:** «Рассмотреть заявку».
  + **Условие/Шлюз (Gateway):** «Решение руководителя?»
    - **[Отклонить]** -> **Действие:** «Указать причину отказа» -> **Действие (в дорожке 'Система'):** «Изменить статус на 'Отклонена'» -> **Действие:** «Уведомить сотрудника» -> **Конечное событие**.
    - **[Одобрить]** -> **Действие (в дорожке 'Система'):** «Изменить статус на 'Одобрена'».
  + Переход в дорожку «HR-специалист».
  + **Действие:** «Внести данные в систему учета».
  + **Действие:** «Обновить график отпусков».
  + **Действие (в дорожке 'Система'):** «Изменить статус на 'Оформлена'» -> **Действие:** «Уведомить сотрудника и руководителя» -> **Конечное событие**.

#### **3. Class Diagram (Диаграмма классов)**

Эта диаграмма описывает основные сущности системы и их взаимосвязи.

* **Классы (Classes):**
  + Пользователь
    - *Атрибуты:* id, ФИО, должность.
    - *Может быть абстрактным классом, от которого наследуются другие.*
  + Сотрудник (наследуется от Пользователь)
    - *Атрибуты:* баланс\_дней\_отпуска.
  + Руководитель (наследуется от Пользователь)
  + HR\_Специалист (наследуется от Пользователь)
  + ЗаявкаНаОтпуск
    - *Атрибуты:* id, дата\_начала, дата\_окончания, статус (enum: На согласовании, Одобрена, Отклонена, Оформлена), причина\_отказа (string, необязательное).

**Система оценивания**

|  |  |
| --- | --- |
| **Критерий** | **Баллы** |
| **1. Use Case Diagram** | 0–4 |
| • 0–1 — отсутствует или неполная модель |  |
| • 2 — частичная модель, не использованы ключевые элементы (шлюзы, роли) |  |
| • 3 — структура верная, но есть незначительные логические ошибки |  |
| • 4 — корректная, логичная BPMN-модель, охвачены все шаги |  |
| **2. Activity Diagram (Диаграмма деятельности)** | 0–2 |
| • 0 — проблемы не указаны или поверхностны |  |
| • 1 — выявлены, но не аргументированы |  |
| • 2 — конкретные проблемы, есть причины и последствия |  |
| **3. Class Diagram (Диаграмма классов)** | 0–3 |
| • 0 — отсутствует или слабо проработан |  |
| • 1 — частично устранены проблемы |  |
| • 2 — логичное улучшение, но без проработки системы |  |
| • 3 — чёткое описание, есть структура, устранены ключевые проблемы |  |
| **4. Разработка KPI** | 0–2 |
| • 0 — отсутствуют |  |
| • 1 — общие, не связанные с процессом |  |
| • 2 — конкретные, измеримые, релевантные процессу |  |

**Максимум: 12 баллов**

**Шкала:**

* **12–10 баллов** — отлично (диаграммы полные, без ошибок, есть оптимизации в VAD).
* **9–7 баллов** — хорошо (незначительные недочёты, например, пропущены 1-2 элемента).
* **6–4 балла** — удовлетворительно (грубые ошибки в нотации или неполные диаграммы).
* **3–0 баллов** — неудовлетворительно (игнорирование ключевых требований).